**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A method of synchronizing data clocked by a first clock to a second clock, comprising:

deriving an offset between the first clock and the second clock; and

fractionally ~~resampling~~ decimating the data during a data overflow, and fractionally interpolating the data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the offset.

2. (Original) The method of claim 1 wherein the data comprises voice.

3. (Original) The method of claim 1 wherein the offset derivation comprises counting at least a portion of a cycle of the first or second clock, the fractional resampling being a function of the count.

4. (Currently Amended) A method of synchronizing data clocked by a first clock to a second clock, comprising:

deriving an offset between the first clock and the second clock; and

fractionally resampling the data as a function of the offset,

wherein the offset derivation comprises counting at least a portion of a cycle of the first or second clock, the fractional resampling being a function of the count; and

~~The method of claim 3~~ wherein the fractional resampling comprises upsampling the data if the count exceeds a threshold and downsampling the data if the count is below the threshold.

5. (Currently Amended) A method of synchronizing data clocked by a first clock to a second clock, comprising:

deriving an offset between the first clock and the second clock;

fractionally resampling the data as a function of the offset; and

~~The method of claim 3 further comprising~~ filtering the count, the data resampling being a function of the filtered count[[.]],

wherein the offset derivation comprises counting at least a portion of a cycle of the first or second clock, the fractional resampling being a function of the count.

6. (Original) The method of claim 1 further comprising receiving the data sampled with the first clock, wherein the offset derivation comprises counting at least a portion of a cycle of the first clock, the fractional resampling of the data being a function of the count.

7. (Original) The method of claim 6 wherein the received data is partitioned into a plurality of frames, and wherein the first clock count comprises incrementing the count at a beginning of each of the frames.

8. (Currently Amended) The method of claim ~~6~~ 5 further comprising generating the second clock by dividing a third clock, wherein the first clock count comprises incrementing the count using the third clock.

9. (Original) The method of claim 8 wherein the fractional resampling comprises resampling the data using the third clock.

10. (Original) The method of claim 1 wherein the offset derivation comprises counting at least a portion of a cycle of the second clock, the fractional resampling of the data being a function of the count, the method further comprising transmitting the fractionally resampled data.

11. (Original) The method of claim 10 wherein the second clock count comprises synchronizing a third clock to the second clock, and counting at least a portion of a cycle of the third clock

12. (Original) The method of claim 10 further comprising generating the second clock by dividing a third clock, wherein the second clock count comprises incrementing the count using the third clock.

13. (Original) The method of claim 12 wherein the fractional resampling comprises resampling the data using the third clock.

14. (Currently Amended) A method of synchronizing data clocked by a first clock to a second clock, comprising:

deriving an offset between the first clock and the second clock; and

fractionally resampling the data as a function of the offset,

~~The method of claim of claim 1~~ wherein the offset derivation comprises counting at least a portion of a cycle of the first clock to produce a first clock count, counting at least a portion of a cycle of the second clock to produce a second clock count, generating an error signal as a function of the first clock count and the second clock ~~counts~~ count, the fractional resampling of the data being a function of the error signal.

15. (Original) The method of claim 14 wherein data is sampled with the first clock, and the fractional resampling comprises upsampling the data when the first clock count exceeds the second clock count, and downsampling the data when the second clock count exceeds the first clock count.

16. (Currently Amended) The method of claim 14 further comprising filtering the first clock count and the second clock ~~counts~~ count to produce a first filtered clock count and a second filtered clock count, the fractional resampling being a function of the ~~filtered counts~~ first filtered clock count and the second filtered clock count.

17. (Currently Amended) A method of synchronizing data exchanged between a cable modem and a cable head end, comprising:

deriving an offset between a cable modem clock and a cable head end clock; and

fractionally ~~resampling~~ decimating the data during a data overflow, and fractionally interpolating the data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the offset.

18. (Original) The method of claim 17 wherein the data comprises voice.

19. (Original) The method of claim 17 wherein the offset derivation comprises counting at least a portion of a cycle of the cable head end clock, the fractional resampling of the data being a function of the count.

20. (Currently Amended) A method of synchronizing data exchanged between a cable modem and a cable head end, comprising:

deriving an offset between a cable modem clock and a cable head end clock;

fractionally resampling the data as a function of the offset; and

~~The method of claim 19 further comprising~~ receiving the data from the cable head end, the data being clocked by the cable head end clock, wherein the fractional resampling comprises upsampling the data if the count exceeds a threshold and downsampling the data if the count is below the threshold[[.]],

wherein the offset derivation comprises counting at least a portion of a cycle of the cable head end clock, the fractional resampling of the data being a function of the count.

21. (Original) The method of claim 19 further comprising clocking the data with the cable modem clock, wherein the fractional resampling comprises downsampling the data if the count exceeds a threshold and upsampling the data if the count is below the threshold, the method further comprising transmitting the fractionally resampled data to the cable head end.

22. (Currently Amended) A method of synchronizing data exchanged between a cable modem and a cable head end, comprising:

deriving an offset between a cable modem clock and a cable head end clock;

fractionally resampling the data as a function of the offset; and

~~The method of claim 19 further comprising~~ filtering the count, the fractional resampling of the data being a function of the filtered count[[.]],

wherein the offset derivation comprises counting at least a portion of a cycle of the cable head end clock, the fractional resampling of the data being a function of the count.

23. (Original) The method of claim 17 further comprising receiving the data from the cable head end, the data being clocked by the cable head end clock, wherein the offset derivation comprises counting at least a portion of a cycle of the cable head end clock, the fractional resampling of the data being a function of the count.

24. (Currently Amended) The method of claim ~~23~~ 22 wherein the received data is partitioned into a plurality of frames, and

wherein the cable head end clock count comprises incrementing the count at a beginning of each of the frames.

25. (Currently Amended) The method of claim ~~23~~ 22 further comprising generating the cable modem clock by dividing a local clock, wherein the cable head end clock count comprises incrementing the count using the local clock.

26. (Original) The method of claim 25 wherein the fractional resampling comprises resampling the data using the local clock.

27. (Original) The method of claim 17 wherein the offset derivation comprises counting at least a portion of a cycle of the cable head end clock, the fractional resampling being a function of the count, the method further comprising transmitting the fractionally resampled data to the cable head end.

28. (Original) The method of claim 27 wherein the cable head end clock count comprises synchronizing a timing recovery clock to the cable head end clock, and counting at least a portion of a cycle of the timing recovery clock

29. (Original) The method of claim 27 further comprising generating the cable modem clock by dividing a local clock, wherein the cable head end clock count comprises incrementing the count using the local clock.

30. (Original) The method of claim 29 wherein the fractional resampling comprises resampling the data using the local clock.

31. (Original) The method of claim of claim 17 wherein the offset derivation comprises counting at least a portion of a cycle of the cable modem clock, counting at least a portion of a cycle of the cable head end clock, generating an error signal as a function of the clock counts, the fractional resampling of data being a function of the error signal.

32. (Original) The method of claim 31 further comprising receiving the data from the cable head end, the data being clocked by the cable head end clock, wherein the fractional resampling comprises upsampling the data when the cable head end clock count exceeds the cable modem clock count, and downsampling the data when the cable modem clock count exceeds the cable head end clock count.

33. (Original) The method of claim 31 further comprising clocking the data with the cable modem clock, wherein the fractional resampling comprises upsampling the data when the cable modem clock count exceeds the cable head end clock count, and downsampling the data when the cable head end clock count exceeds the cable modem clock count.

34. (Original) The method of claim 31 further comprising filtering the cable modem and cable head end clock counts, the fractional resampling being a function of the filtered counts.

35. (Original) The method of claim 31 further comprising receiving the data from the cable head end, the data being partitioned into a plurality of frames, wherein the cable

head end clock count comprises incrementing the count at a beginning of each of the frames.

36. (Original) The method of claim 31 wherein the cable head end clock count comprises synchronizing a timing recovery clock to the cable head end clock, and counting at least a portion of a cycle of the timing recovery clock

37. (Currently Amended) A synchronization circuit, comprising:

a timing recovery clock adapted to be synchronized by an external source;

a counter to count at least a portion of a cycle of the timing recovery clock; and

a sample tracker adapted to receive sampled data, the sample tracker fractionally ~~resampling~~ decimating the sampled data during a data overflow, and fractionally interpolating the data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the count.

38. (Currently Amended) A synchronization circuit, comprising:

a timing recovery clock adapted to be synchronized by an external source;

a counter to count at least a portion of a cycle of the timing recovery clock; and

a sample tracker adapted to receive sampled data, the sample tracker fractionally resampling the sampled data as a function of the count,

~~The synchronization circuit of claim 37~~ wherein the sample tracker downsamples the data if the count exceeds a threshold and upsamples the data if the count is below the threshold.

39. (Currently Amended) A synchronization circuit, comprising:

a timing recovery clock adapted to be synchronized by an external source;

a counter to count at least a portion of a cycle of the timing recovery clock;

a sample tracker adapted to receive sampled data, the sample tracker fractionally resampling the sampled data as a function of the count; and

~~The synchronization circuit of claim 37 further comprising~~ a filter between the counter and the sample tracker.

40. (Original) The synchronization circuit of claim 37 further comprising an analog-to-digital converter to generate the sampled data from an analog voice signal, an analog-to-digital clock to clock the counter, and a clock divider to divide the analog-to-digital clock, the analog-to-digital converter being clocked by the divided analog-to-digital converter clock.

41. (Original) The synchronization circuit of claim 40 further comprising a voice queue to store the resampled data.

42. (Currently Amended) A synchronization circuit, comprising:

a sample tracker to receive a plurality of frames of sampled data from an external source; and

a counter to count the frames of the sampled data, wherein the sample tracker fractionally ~~resamples~~ decimates the sampled data during a data overflow, and fractionally interpolates the data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the count.

43. (Currently Amended) A synchronization circuit, comprising:

a sample tracker to receive a plurality of frames of sampled data from an external source; and

a counter to count the frames of the sampled data, wherein the sample tracker fractionally resamples the sampled data as a function of the count, and

~~The synchronization circuit of claim 42~~ wherein the sample tracker upsamples the data if the count exceeds a threshold and downsamples the data if the count is below the threshold.

44. (Original) The synchronization circuit of claim 42 further comprising filter between the counter and the sample tracker.

45. (Original) The synchronization circuit of claim 42 further comprising an digital-to-analog to convert the fractionally resampled data to an analog voice signal, a digital-to-analog clock to clock the counter, and a clock divider to divide the digital-to-analog clock, the digital-to-analog converter being clocked by the divided digital-to-analog converter clock.

46. (Original) The synchronization circuit of claim 42 further comprising a voice queue to buffer the frames of sampled

data, the sample tracker fractionally resampling the buffered sampled data.

47. (Original) The synchronization circuit of claim 46 further comprising a processor to increment the counter every time a frame of sampled data is received by the voice queue.

48. (Currently Amended) A synchronization circuit, comprising:

a first counter to count at least a portion of a cycle of a first clock;

a second counter to count at least a portion of a cycle of a second clock; and

a sample tracker adapted to receive sampled data, the sample tracker fractionally ~~resampling~~ decimating the sampled data during a data overflow, and fractionally interpolating the data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the error signal.

49. (Currently Amended) A synchronization circuit, comprising:

a first counter to count at least a portion of a cycle of a first clock;

a second counter to count at least a portion of a cycle of a second clock; and

a sample tracker adapted to receive sampled data, the sample tracker fractionally resampling the sampled data as a function of the error signal,

~~The synchronization circuit of claim 48~~ wherein the data is sampled with the first clock, and sample tracker upsamples the data when the first clock count exceeds the second clock count,

and downsamples the data when the second clock count exceeds the first clock count.

50. (Currently Amended) A synchronization circuit, comprising:

a first counter to count at least a portion of a cycle of a first clock;

a second counter to count at least a portion of a cycle of a second clock;

a sample tracker adapted to receive sampled data, the sample tracker fractionally resampling the sampled data as a function of the error signal; and

~~The synchronization circuit of claim 48 further comprising~~ a first filter between the first counter and the sample tracker, and a second filter between the second counter and the sample tracker.

51. (Original) The synchronization circuit of claim 48 wherein the data is partitioned into a plurality of frames and sampled with the first clock, and wherein the first counter is incremented with each of the frames of the sampled data.

52. (Original) The synchronization circuit of claim 48 wherein the data is sampled with the first clock, the synchronizing circuit further comprising a timing recover clock synchronized to the second clock, the second counter being responsive to the timing recovery clock.

53. (Currently Amended) A synchronization circuit, comprising:

error means for generating an error signal as a function of an offset between a first clock and a second clock; and

~~resampling~~ decimating means, adapted to receive sampled data, for fractionally ~~resampling~~ decimating the sampled data during a data overflow, and interpolating means, adapted to receive sampled data, for fractionally interpolating the sampled data during a data underflow,

wherein the data overflow and the data underflow are determined as a function of the error signal.

54. (Original) The synchronization circuit of claim 53 wherein the error means comprises counting means for counting at least a portion of a cycle of the first or second clock, the resampling means fractionally resampling the sampled data as a function of the count.

55. (Currently Amended) A synchronization circuit, comprising:

error means for generating an error signal as a function of an offset between a first clock and a second clock; and

resampling means, adapted to receive sampled data, for fractionally resampling the sampled data as a function of the error signal,

wherein the error means comprises counting means for counting at least a portion of a cycle of the first or second clock, the resampling means fractionally resampling the sampled data as a function of the count, and

~~The synchronization circuit of claim 54~~ wherein the resampling means further comprises means for upsampling the data if the count exceeds a threshold and means for downsampling the data if the count is below the threshold.

56. (Currently Amended) A synchronization circuit, comprising:

error means for generating an error signal as a function of an offset between a first clock and a second clock;

resampling means, adapted to receive sampled data, for fractionally resampling the sampled data as a function of the error signal; and

~~The synchronization circuit of claim 54 further comprising~~ filtering means for filtering the count, the resampling means fractionally resampling the sampled data as a function of the filtered count[[.]],

wherein the error means comprises counting means for counting at least a portion of a cycle of the first or second clock, the resampling means fractionally resampling the sampled data as a function of the count.

57. (Original) The synchronization circuit of claim 53 wherein the data is sampled with the first clock, the error means comprises counting means for counting at least a portion of a cycle of the first clock, the fractional resampling of the sampled data being a function of the count.

58. (Original) The synchronization circuit of claim 57 wherein the data is partitioned into a plurality of frames, and wherein the counting means is incremented at a beginning of each of the frames.

59. (Original) The synchronization circuit of claim 57 further comprising means for generating the second clock by dividing a third clock, wherein the counting means is incremented using the third clock.

60. (Original) The synchronization circuit of claim 59 wherein the resampling means fractionally resamples the sampled data with the third clock.

61. (Original) The synchronization circuit of claim 53 wherein the error means comprises counting means for counting at least a portion of a cycle of the second clock, the resampling means fractionally resampling the data as a function of the count.

62. (Original) The synchronization circuit of claim 61 wherein the error means further comprises means for synchronizing a third clock to the second clock, the counting means counting at least a portion of a cycle of the third clock

63. (Original) The synchronization circuit of claim 61 further comprising means for generating the second clock by dividing a third clock, the counting means being incremented using the third clock.

64. (Original) The synchronization circuit of claim 63 wherein the resampling means fractionally resamples the sampled data with the third clock.

65. (Original) The synchronization circuit of claim of claim 53 wherein the error means comprises first counting means for counting at least a portion of a cycle of the first clock, second counting means for counting at least a portion of a cycle of the second clock, the error signal being a function of the first and second clock counts, and the resampling means fractionally resampling of the sampled data as a function of the error signal.

66. (Original) The synchronization circuit of claim 65 wherein data is sampled with the first clock, and the resampling means comprising means for upsampling the sampled data when the first clock count exceeds the second clock count, and means for downsampling the data when the second clock count exceeds the first clock count.

67. (Original) The synchronization circuit of claim 65 further comprising filtering means for filtering the first and second clock counts, the resampling means fractionally resampling the sampled data as a function of the filtered counts.